

LONGEST COMMON

SUBSEQUENCE

```
class Solution {  
    public int longestCommonSubsequence(String text1, String text2) {  
        int[][] sequenceLengths = new int[text1.length() + 1][text2.length() + 1];  
        // Fill top row  
        for (int i = 0; i < sequenceLengths[0].length; i++)  
            sequenceLengths[0][i] = 0;  
        // Fill top col  
        for (int i = 0; i < sequenceLengths.length; i++)  
            sequenceLengths[i][0] = 0;  
        // Fill in the middle cells  
        for (int i = 1; i < text1.length() + 1; i++)  
            for (int j = 1; j < text2.length() + 1; j++)  
                if (text1.charAt(i - 1) == text2.charAt(j - 1))  
                    sequenceLengths[i][j] = sequenceLengths[i - 1][j - 1] + 1;  
                else {  
                    // If immediate top is greater than immediate left  
                    // otherwise take immediate left  
                    sequenceLengths[i][j] = Math.max(sequenceLengths[i - 1][j], sequenceLengths[i][j - 1]);  
                }  
        }  
        return sequenceLengths[text1.length()][text2.length()];  
    }  
}
```

str1 = ABCD

		A	B	C	D
A		1	1	1	1
C		1	1	2	2
A		1	2	2	2
B		1	2	2	2
D		1	1	1	2

str2 = CABD

